ESD TDR 64-618

ESTI FILE COPY

ESD-TDR-64-618

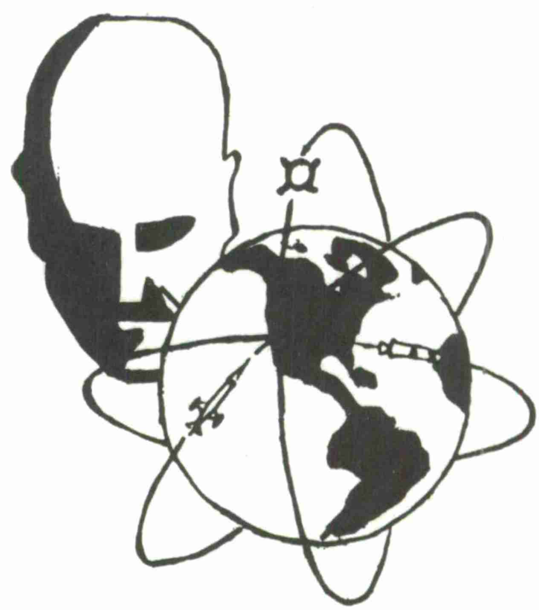
PRELIMINARY RESEARCH ON THE TAXONOMY OF SUBJECT MATTER

TECHNICAL DOCUMENTARY REPORT NO. ESD-TDR-64-618

NOVEMBER 1964

T. K. Roderburg
H. D. Cluck
G. R. Murray, Jr.

DECISION SCIENCES LABORATORY
ELECTRONIC SYSTEMS DIVISION
AIR FORCE SYSTEMS COMMAND
UNITED STATES AIR FORCE
L. G. Hanscom Field, Bedford, Massachusetts



Project 7682, Task 768204

(Prepared under Contract No. AF 19 (628)-2407 by the Operations Research Center, Massachusetts Institute of Technology, Cambridge, Massachusetts.)

ESD-TDR-64-618

PRELIMINARY RESEARCH ON THE TAXONOMY OF SUBJECT MATTER

TECHNICAL DOCUMENTARY REPORT NO. ESD-TDR-64-618

NOVEMBER 1964

T. K. Roderburg
H. D. Cluck
G. R. Murray, Jr.

DECISION SCIENCES LABORATORY
ELECTRONIC SYSTEMS DIVISION
AIR FORCE SYSTEMS COMMAND
UNITED STATES AIR FORCE
L. G. Hanscom Field, Bedford, Massachusetts

Project 7682, Task 768204

(Prepared under Contract No. AF 19 (628)-2407 by the Operations Research Center, Massachusetts Institute of Technology, Cambridge, Massachusetts.)

ESD-TDR-64-618

FOREWORD

This research was conducted under Contract AF 19(628)-2407 in support of Project 7682, Man-Computer Information Processing, Task 768204, Automated Training for Information Systems. The authors were Messrs. T. K. Roderburg, H. D. Cluck, and Dr. G. R. Murray, Jr., of Massachusetts Institute of Technology. The Air Force Technical Monitor was Dr. Sylvia R. Mayer of the Decision Sciences Laboratory.

This report, "Preliminary Research on the Taxonomy of Subject Matter," has been included in the series of technical notes of the Operations Research Center, M. I. T., and has been assigned for this purpose the secondary report number "Technical Note No. 3, Operations Research Center, M. I. T."

REVIEW AND APPROVAL

This technical report has been reviewed and is approved.

Joseph T. Begley
JOSEPH T. BEGLEY
Chief, Applications Division
Decision Sciences Laboratory

for James S. Duva
ROY MORGAN
Colonel, USAF
Director, Decision Sciences Laboratory

ESD-TDR- 64-618

PRELIMINARY RESEARCH ON THE TAXONOMY OF SUBJECT MATTER

ABSTRACT

Preliminary research in the taxonomy of subject matter is reported. This work is part of a program of study aimed at developing computational methods useful in the preparation of educational matter for presentation by machine. Basic concepts of subject structure are defined. A language processing program that assists the classification of subject matter is described and its use illustrated. An experiment on the variations in subject structure as seen by different individuals is reported.

KEY WORD LIST

1. Classification
2. Data Processing Systems
3. Teaching Machines
4. Computers
5. Language
6. Machine Translation
7. Programming
8. Experimental Data
9. Training Material

TABLE OF CONTENTS

LIST OF ILLUSTRATIONS

PRELIMINARY RESEARCH ON THE TAXONOMY OF SUBJECT MATTER

1. Introduction

The preparation of teaching material for programmed teaching machines is a complex task. Computer-directed teaching machines require that a structure be identified for each subject taught[1]. The research reported here is an attempt to find ways in which the computer itself may be employed to simplify the job of teaching program preparation. Although the work performed so far represents only a meager step in the desired direction, several results have emerged from this work that are noteworthy in themselves and may be useful in establishing the types of teaching that can be profitably carried out by machine. It is for this reason that we have prepared this report on our very preliminary findings.

2. The Precedence Graph

In the preparation of programs for sophisticated computer-controlled teaching machines, there must be precedence relationships between the individual elements of information used in the instruction of a subject. We say that Element A precedes Element B if the knowledge of A is required for the understanding of B.

The precedence of relationship may conveniently be given by a graph with directional branches where the nodes of the graph

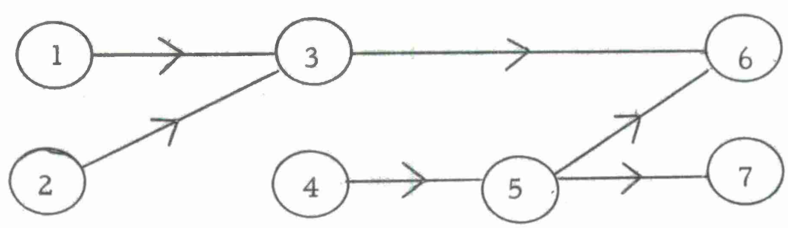


Figure 2.1 A Precedence Graph

represent the individual elements of information, and by convention, a node (element) is preceded by all nodes (elements) lying on paths leading to the node in question. Such a graph will be called a precedence graph. In Figure 2.1 Node 6 is, for instance, preceded by all the nodes with lower numbers (1 - 5), while Node 7 is preceded only by 4 and 5.

Possible uses of the precedence relationships or the precedence graphs include the following:

a) A teaching machine may be programmed to make decisions regarding the mode of presentation of one or more succeeding elements where the node selected must depend upon the student's previous performance. Clearly, the performance on preceding elements (in the particular sense used here) is most relevant for selection of the best model.

b) The precedence graph indicates the best order of presenting elements. In general, where more than one path leads to a node it will be advantageous to present those elements represented by the longest paths first.

c) The precedence graph indicates the minimum number of elements required for the understanding of any particular element or set of elements.

d) If the student fails in comprehending a particular element, the precedence graph indicates which elements should be considered in the remedial instruction.

3. Procedure for Establishing Precedence Graphs

In order to prepare the precedence graph the writer of the teaching program (the instructor) may start with all the elements in some natural order of presentation, e.g. in the order in which they appear in a text-book on the subject. Generally, there will be a number of possible such arrangements of the material, all having the property that for any element, all preceding elements appear ahead of it. (The finished precedence graph will in fact indicate the various possible orders of presentation with the given division of the subject matter into individual elements.) The instructor first prepares a list giving each element an identifying number according to the initial order of presentation (the first being designated 1, the second 2, etc.). The list could also contain an abbreviated statement of the contents of each element. On the basis of this list and his knowledge of the subject it is a reasonably easy task for the instructor to write down, for each element in turn, the elements which immediately precede the one being considered. By definition Element A immediately precedes Element B, if on the precedence graph there is a branch from A to B and no other paths between A and B (if other paths exist, the branch A-B is redundant.) Thus, in Figure 2.1 nodes 3 and 5 are immediately preceding 6, while 7 is immediately preceded by 5 only. If the subject contains a large number of elements in complicated precedence relationships, it is possible that the instructor may miss some immediately preceding elements and, in some cases, may indicate elements as immediately preceding when in fact they are not. The method described below for constructing the graph allows the possibility of errors in the original list. It provides aids for checking the data and (ultimately) freeing it of errors.

The list of elements with their immediate predecessors permit construction of the precedence graph. If the instructor has missed some immediately preceding elements, branches will be missing and if he has included elements on the list which are not immediately preceding, there will be redundant paths on the graph. If, for instance, the instructor had mistakenly listed Element 2 as immediately preceding Element 6, this would have produced a diagram like Figure 2.1 but containing a redundant branch between nodes 2 and 6. The path is redundant because 2 is already preceding 6 by virtue of the path 2-3-6.

Depending upon the complexity of the precedence graph, redundant paths may be fairly easy to spot. In order to eliminate the possibility of missing paths, however, it is necessary for the instructor to decide for each element which lower-numbered elements are not on paths leading to it, and then decide whether any of these should actually precede the element under consideration.

A computer program for the IBM 7094 has been written to assist the instructor in the preparation of the precedence graph. A brief description of the function of this program is included here; the following section describes the action and use of the program in greater detail for those interested in employing the program in their work.

The input to the program is the instructor's list of basic elements contained in the subject, numbered approximately in their order of appearance in a logical presentation. The diagram (Fig. 3.1) and the following discussion use the symbol E to represent this list of basic elements. A second part of the input is the instructor's assignment to each basic element of the immediately preceding element (if any). The symbol IPE is used below to represent this second list. The function of the computer program is to eliminate members of the list IPE that produce redundant paths in the graph. The output of the program is the list of basic elements, E, the revised list of immediately preceding elements, IPE, and a list that shows the basic elements of lower number that do not fall on any path leading up to each basic element. This final list, called NPE, is reviewed by the instructor to determine whether any of his original assignments on the list IPE are in error.

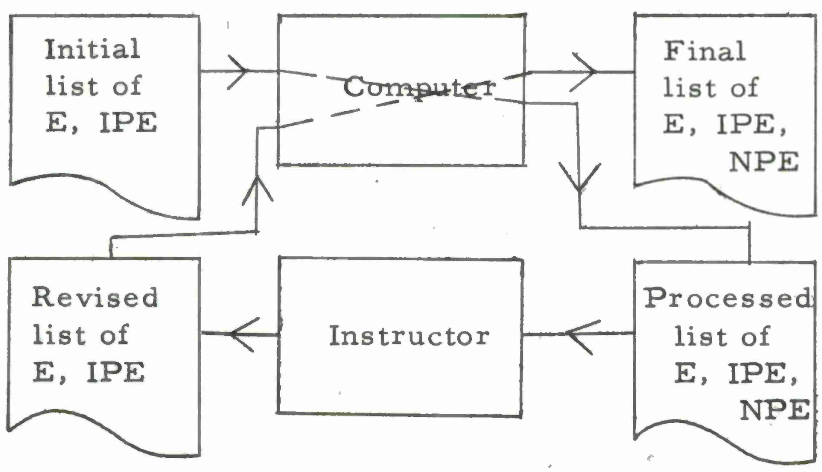


Figure 3.1 - Flow Diagram of Computer-Aided Revision of Precedence List

4. Use of the Computer Program

For programming the processing of the precedence relations it is convenient to use list-processing language (2) which permits manipulation of symbols having non-numerical meaning and avoids the necessity for specifying the storage requirements in advance. These programs are especially designed for complex, non-numeric problems, and permit the use of recursive functions.

The language used for programming this type of problem is LISP (2). In the LISP language all the data are in the form of symbolic expressions of indefinite length. The program is shown in Appendix A. The expressions shown may be punched consecutively on cards. In addition to these cards, an FMS ID-card, an XEQ-card, binary deck LSP CAL 00-09, DATA-card, and binary card TRA 176 are required. These additional cards, available from the dispatcher at the M. I. T. Computation Center, should be placed in front of the program cards. The input data, i.e. the list of immediately preceding elements (IPE), are contained within the double set of parentheses after SETUP (line 5 from the bottom). The IPE are listed for each element in sequence starting with Element 1. The IPE for each element are listed within parentheses in ascending order, each IPE being separated by a space only. When an element has no IPE, the symbol NIL is used without parentheses. In addition to the list, the total number of nodes or elements must be given in the parentheses after JOB 1 and NOT BEFORE (lines 3 and 4 from the bottom). The number after NODES (line 1 from the top) must always be greater than the number of nodes.

The program produces two output lists with the following formats:

a) Under the heading JOB 1:

(The total number of elements) Element No. (IPE of element on preceding line)	Repeated as required

b) Under the heading NOT BEFORE:

(The total number of elements)
(NPE of last element on line) } Repeated as required

5. An Example

Suppose a list of IPE is produced as shown in Table 5.1.

Element No.	IPE
1	None
2	1
3	None
4	2, 3
5	4
6	4
7	4, 5, 6
8	2, 7
9	7
10	3, 6, 7

Table 5.1 - Input List of IPE

The corresponding precedence graph may easily be constructed from the information in Table 5.1, as shown in Figure 5.1.

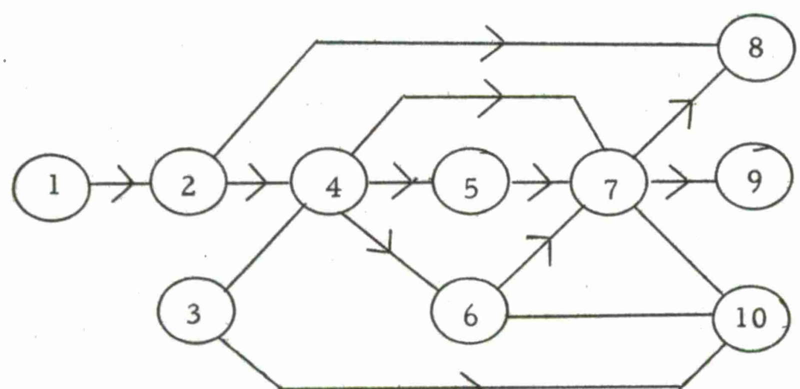


Figure 5.1 - Initial Precedence Graph

The graph obviously has a number of redundant paths. With a problem of this magnitude it is easy to eliminate the redundant paths by inspection. However, if the computer-aided process is used, the input list after SETUP becomes:

((NIL) (1) NIL (2 3) (4) (4) (4 5 6) (2 7) (7) (3 6 7))) and the output is:

JOB 1	NOT BEFORE
(10)	(10)
10	(8 9 10)
(7)	
9	(8 9)
(7)	
8	(8)
(7)	
7	(7)
(5 6)	
6	(5 6)
(4)	
5	(5)
(4)	
4	(4)
(2 3)	
3	(1 2 3)
NIL	
2	(2)
(1)	
1	(1)
NIL	

The lists JOB 1 and NOT BEFORE actually appear under each other in the printed output, but are shown side by side above in order to save space. From the list JOB 1 one may easily produce the corrected precedence graph shown in Figure 5. 2.

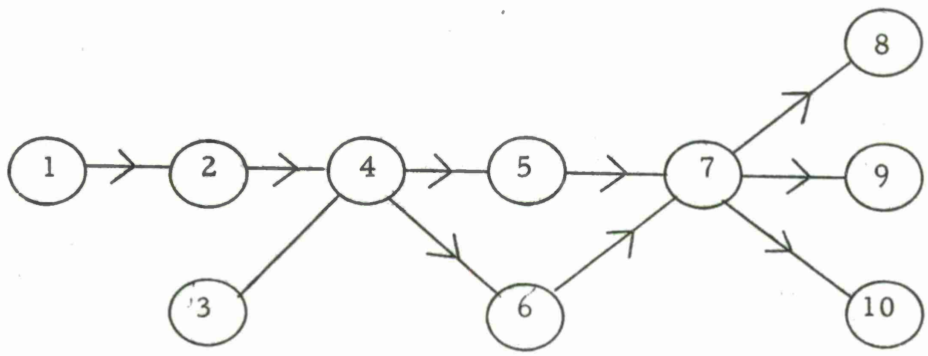


Figure 5. 2 - Corrected Precedence Graph

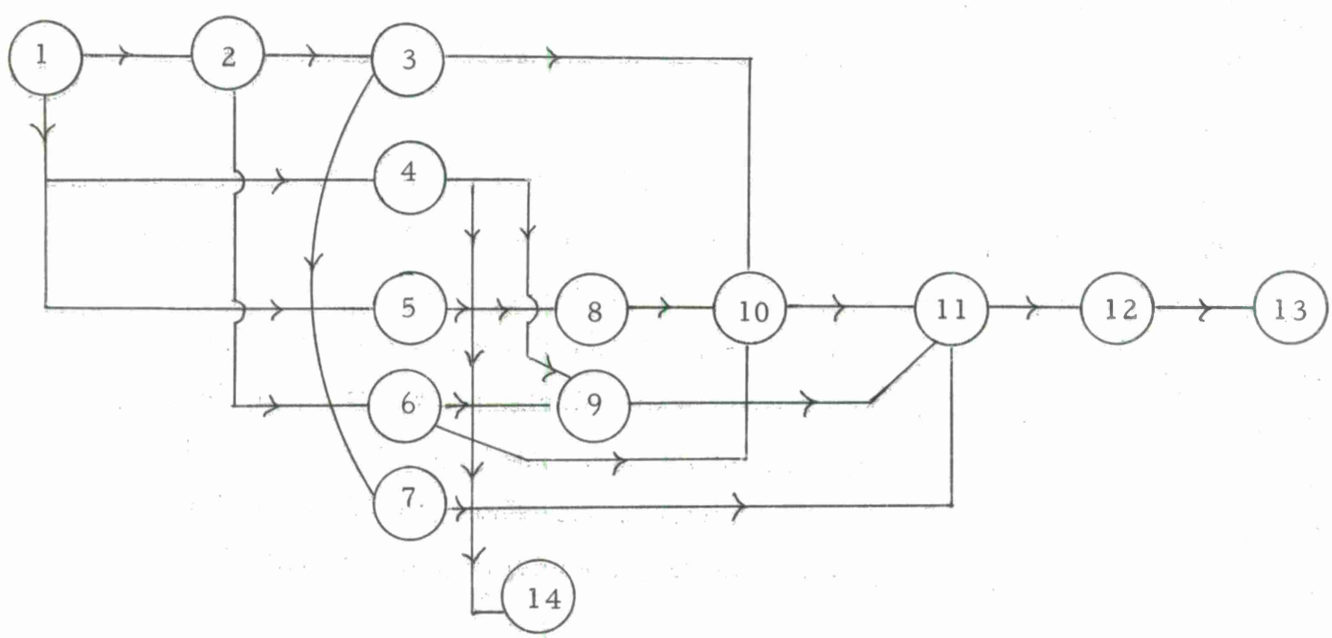
6. Application to Miniature Geometry

Smallwood[1] prepared a computer-directed teaching machine for the subject of miniature geometry. The subject matter is that covered on pages 115-119 of "Adaptive Teaching Machine Program for Miniature Geometry"[4]. This same material has been subjected to the analysis described in Section 2, above. The subject was decomposed (by the original authors) into a paragraph of introductory concepts, two formal definitions, four axioms, six theorems, and a final statement on the independence of axioms. Study of the material resulted in the initial assignment shown in Table 6. 1. Application of the program described in Section 3 above yields the final precedence graph shown in Figure 6. 1.

Table 6.1

Decomposition of Miniature Geometry

1.	Introduction of the terms "point" and "line" and their naming	
2.	Definition 1	(1)
3.	Definition 2	(2)
4.	Axiom 1	(1)
5.	Axiom 2	(1)
6.	Axiom 3	(2)
7.	Axiom 4	(2, 3)
8.	Theorem 1	(5)
9.	Theorem 2	(6, 4)
10.	Theorem 3	(8, 6, 5, 3)
11.	Theorem 4	(10, 9, 7)
12.	Theorem 5	(11, 6, 5)
13.	Theorem 6	(12, 11, 10, 7, 6, 5, 3)
14.	Independence of Axioms	(7, 6, 5, 4)



7. Experiments Performed

An experiment was conducted to determine the degree to which precedence graphs of different individuals might differ. The first chapter of Determinants and Matrices by A. C. Aitken (5) was organized into basic elements. Eleven M.I.T. graduate students, all generally familiar with matrix algebra, were asked individually to study the list of basic elements and to identify the immediately preceding element for each element in the basic list.

The instructions presented to the eleven students, and the answers obtained from their work are summarized in Appendix B.

The results of this experiment show that there is a substantial variation among the graphs. Before examining the details of the results we should recognize that there may be several reasons for this variation. Some students may have understood the material better than others. Some may have worked more carefully. In some cases the meaning of two paragraphs (each considered to be an element) may be very similar and the choice of the proper one becomes an arbitrary act. Sometimes both paragraphs will be chosen.

Some examples of the latter cause of variation can be seen in the results of this experiment. Element 6 was said to be dependent upon Element 4 by 4 students and dependent upon Element 5 by 5 students. Because all of the students who said that Element 6 was dependent upon Element 5 also said that Element 5 was dependent upon Element 4, they were indirectly saying that Element 6 was dependent upon both elements 4 and 5. The same situation is encountered with Element 27 (dependent upon elements 25 and 26) and Element 32 (dependent upon elements 30 and 31).

In order to evaluate the results, we need some sort of measurement of the degree to which the graphs differ. Three types of measurements were investigated.

Table 7.1 contains the tabulation of the number of times an element was said to be required to understand some other given element. The table is simply the sum of the charts prepared by the students, summing together all corresponding boxes. This table shows clearly the extent of the variation of the individual graphs. From such a table one might construct a partial precedence graph by taking the decision of the majority of the students. Such a graph is shown in Figure 7.1 (the dashed lines indicate the two situations where there was not a majority agreement but where the majority was divided between two elements whose content of information might be very similar. In each case, one of the two paths would be correct, but not both). This partial graph might be considered as the "backbone" framework for the complete graph. Whether this would truly be the correct backbone framework is unknown since we do not have a "correct" precedence graph with which to judge the experimental results. Perhaps instead of taking the decision of a simple majority, we could take the decisions agreed upon by some higher percentage. The higher the percentage used, the closer would come agreement with the "correct" graph.

Another measurement considered was the degree of agreement as to which elements were most important. By most important we mean those elements most often required to understand other elements. Table 7.2 lists those elements which each student thought were most important. In this case most important was taken to mean that the elements were required at least three times. Table 7.3 lists the frequency with which each element was considered

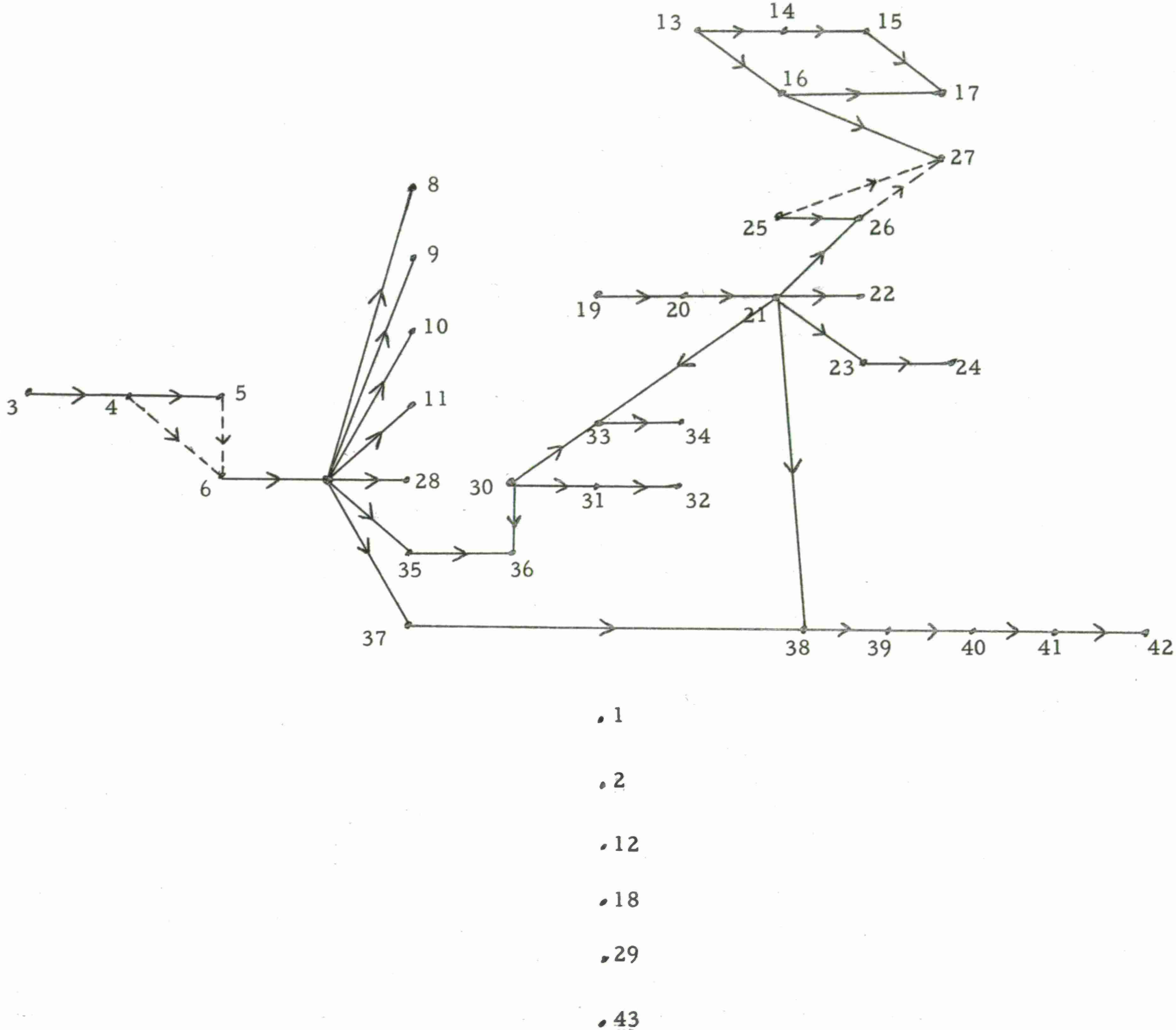


Figure 7.1 - Partial Precedence Graph (Majority Decision)

Table 7.2

List of "Important" Elements

Student Number	Important Elements
1	7, 16, 21, 25, 30
2	7, 12, 14, 21, 23, 30
3	7, 21, 30
4	7, 11, 21, 30
5	2, 7, 14, 21, 30
6	11, 21
7	7, 14, 23
8	11, 29
9	4, 7, 12, 15, 21, 30
10	7, 12, 14, 16, 21, 25
11	7, 21, 25, 30

Table 7.3

Frequency of "Important" Elements

Element	No. of Students Who Considered It Important
2	1
4	1
7	9
11	3
12	3
14	4
15	1
16	2
21	9
23	2
25	3
29	1
30	7

Table 7.4

List of Inputs and Outputs

Student No.	No. Inputs To Graph	No. Outputs To Graph
1	5	21
2	3	14
3	7	17
4	1	10
5	6	20
6	4	8
7	4	16
8	1	7
9	1	17
10	4	17
11	5	20

important. There was strong agreement on the importance of only three elements: 7, 21 and 30. All the other elements considered important were so selected by a minority. If the same definition of important is applied to the partial precedence graph in Figure 1, the same results are obtained; that elements 7, 21 and 30 are the most important.

Another measurement suggested was to count the number of inputs and outputs of each graph. An input is an element that is not dependent upon any other element. An output is an element which is not required (is not depended upon) by some other element. Table 7.4 lists these for the experimental graphs. About the only significant conclusion is that there are more outputs than inputs, the ratio in most cases being from 3 to 5.

From the instructions it is seen that the experimental graphs were going to be processed by a computer program. This program was to eliminate redundancies and list the non-dependent elements for each given element. The processed graphs were then to be returned to the students for their review to determine whether they had "overlooked" any dependent elements. However, because of the little time remaining in the school term during which this work was done, and thus the little time available for the students to review the processed graphs, the computer processing was not done. Rather, redundancies in the graphs were determined manually. Because of the relatively small number of elements and the low degree of inter-dependency, this was not difficult. Five of the graphs contained redundancies.

8. Conclusions

The graphs shown in Figure 6.1 and 7.1 have the property that a particular element in the material may be reached, in almost all cases, by a path that contains only a small fraction of the entire set of elements. Thus if one seeks to understand only one result in either of these subjects (Miniature Geometry and Matrix Theory) it is necessary only to learn a small fraction of the subject. The experience summarized in figures 6.1 and 7.1 is admittedly very small. If we imagine, for the moment, that these results are typical, then we see that computer-directed teaching may be extremely helpful to the student with specific needs, a task for which a textbook is not well suited. The decomposition of a subject into its basic elements and the establishment of precedence relations among these elements depend to some extent on the individual who has undertaken these tasks. It is difficult to draw very definite conclusions from the experiments performed so far. One troublesome feature of this work is the difficulty in arriving at a standard measure of the difference between two precedence graphs for the same subject. Despite the differences in detail among the graphs of Matrix Theory prepared by the eleven students, it was still possible to identify a reasonable "concensus" graph.

APPENDIX A

LISP PROGRAM FOR PROCESSING PRECEDENCE LISTS

```
TEST
ARRAY(((NODES(100)LIST)))
DEFINE((
(SETUP(LAMBDA(Y)(PROG (N V )
(SETQ V Y)
(SETQ N 1)
A (COND((NULL V)(RETURN NIL)))
(NODES(QUOTE SET)(CAR V) N)
(SETQ N(ADD1 N))
(SETQ V (CDR V))
(GOA)  )))
(JOB1(LAMBDA(Y)(PROG(N V)
(SETQ N Y)
A  (SETQ V(REMOVE(NODES V)))
(PRINT N)
(NODES(QUOTE SET) V N)
(PRINT(NODES N))
(SETQ N(SUB1 N))
(COND((ZEROP N)(RETURN NIL)))
(GO A)  )))
(REMOVE(LAMBDA(X)(COND
((NULL X)NIL)
((REMOVE1(CAR X)(CDR X))(REMOVE(CDR X)))
(T(CONS(CAR X)(REMOVE(CDR X))))  )))
(REMOVE 1(LAMBDA(X Y)(COND
((NULL Y)NIL)
(T(OR(EQUAL X(CAR Y))(REMOVE1 X(NODES(CAR Y)))
(REMOVE1 X(CDR Y))))  )))
(NOTBEFORE(LAMBDA(Y)(PROG(N V)
(SETQ N Y)
A  (SETQ V (BEFORE(NODES N)))
(PRINT(NOTBEFORE1  V N))
(SETQ N(SUB1 N))
(COND((ZEROP N)(RETURN NIL)))
(GO A)  )))
(BEFORE(LAMBDA(Y)(COND
(( NULL Y)NIL)
(T(CONS(CAR Y)(APPEND(BEFORE(NODES(CAR Y)))(BEFORE(CDR Y)))))
)))
(NOTBEFORE1(LAMBDA(X N)(PROG(V U)
```

```
(  SET Q V N)
A (COND((NOT(MEMBER V X))(SETQ U(CONS V U))))
(SETQ V (SUBI V))
(COND((ZEROP V)(RETURN U)))
(GO A)  )))
))
EXCISE(*T*)
SETUP ((List of immediately preceding element*))
JOB (Number of elements **)
NOTBEFORE (Number of elements**)
```

---

*E. g. , if element 1 is not preceded by any, element 2 is preceded by 1, and element 3 is immediately preceded by 2 and 1, the entry in this place becomes: NIL (1) (2 3)

**E. g. , if the total number of elements is three, the entry is 3.

APPENDIX B

PRECEDENCE GRAPH EXPERIMENT

An experiment is being conducted to determine the variability of Precedence Graph construction among a selected group of individuals. Your participation in this experiment would be appreciated. The instructions for your part of the experiment will be preceded by an explanation of what Precedence Graphs are and to what use they may be put.

Precedence Graphs

Every book or article read contains a set of ideas or concepts which the author wishes to convey to you. These ideas and concepts might be better described as elements of information. A book then could be described as a set of information elements, or more briefly, as a set of elements.

The ability of the reader to understand the significance of each element is dependent upon some amount of prior knowledge. Some prior knowledge is necessary before any element in the set can be understood. This is the background that the author assumes his readers to have. In addition, the understanding of certain elements will require the prior understanding of other elements in the same set. Thus the elements in a set could be ordered according to the order in which they can be understood. In general there can be many such orderings of the set. For instance, there will be some elements which are not dependent upon any of the other elements in the set. Any one of these could be the first element in an acceptable ordering.

The simple ordering of a set is unsatisfactory in describing a set since it does not reveal all the interrelationships between the elements. A more complete description is obtained by listing all the elements which must precede each element in the set. Figure 1 shows how such a listing could be presented graphically. The elements have been assigned numbers in a manner such that an element is never preceded by an element of a higher number. The arrows on the branches indicate the direction representing a higher level of understanding.

Element	Preceded by
1	none
2	none
3	1, 2
4	none
5	4
6	1, 2, 3, 4, 5
7	4, 5

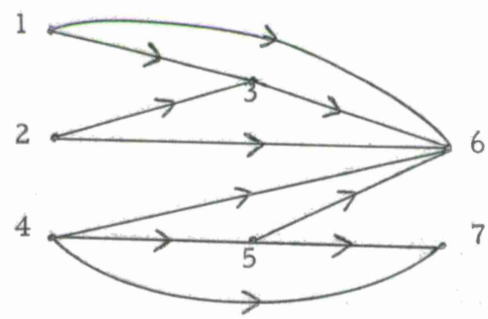
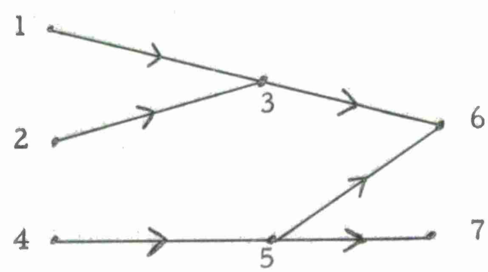


Figure 1

The graph in Figure 1 contains unnecessary (redundant) branches. For instance, element 6 must be preceded by elements 1, 2, 3, 4 and 5. However element 3 is preceded by elements 1 and 2 and element 5 by element 4. Thus if elements 3 and 5 are understood, element 6 can be understood. A similar situation exists for element 7. Thus the graph could be simplified and redrawn as shown in Figure 2. Figure 2 will be referred to as a Precedence Graph.



Precedence Graph

Figure 2

Whenever the author writes his book, he must be aware of the Precedence Graph for his set of elements. He would not present element 6 (referring to Figure 2) before presenting element 1. If he did, he might sell more books by advertising it as a mystery story. He does see however that there are many orderings in which he might present his set of elements. By examining the characteristics of his expected reorders, he selects the ordering which he believes will give the greatest satisfaction to his readers.

A teaching machine, unlike the author, need not decide beforehand the exact ordering of the elements. Depending upon the background of the student confronting the teaching machine at any given moment, the teaching machine may adjust its presentation. Furthermore, as the student progresses through the elements, answering questions at various points, the teaching machine can further adjust its presentation. If the student fails to understand some of the elements, the teaching machine can take remedial action by repeating the presentation of the pertinent elements. If the student only wants to learn about a few of the elements in the set, the teaching machine can again select the proper presentation. To have these capabilities, it must have the Precedence Graph programmed into it.

Experiment

The present experiment is to determine how the Precedence Graphs, constructed for a set of elements by different individuals, vary. Attached to these instructions is a copy of the first chapter in Determinants and Matrices by A. C. Aitken[5]. The author's presentation represents just one possible ordering of the information elements. You are to construct the Precedence Graph from which all the possible orderings can be determined.

In order to standardize the numbering of the elements, each major paragraph has been assigned a number. Thus, in this case, an element (a major paragraph) may contain several ideas or concepts. Your results are to be tabulated on the attached answer sheet.

Two types of errors can be made in constructing the Precedence Graph; unnecessary branches (redundancy) may not be removed and precedence relationships may be overlooked. A computer program has been written to facilitate the discovery and correction of these errors. The computer program automatically removes the unnecessary branches. It also provides a list of non-preceding lower-numbered elements for each element. This list can then be examined to determine whether any preceding elements in fact have been overlooked. Your Precedence Graph will be checked with this program and the results returned to you for your examination and correction.

Non-Redundant Preceding Elements

Element	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42
2	4																																									
3	2	4																																								
4	1	5	7																																							
5				10																																						
6		1	1	4	5																																					
7				1		10																																				
8	1					1	8																																			
9						2	8	1																																		
10						1	7		1																																	
11						1	7	2	1	1																																
12						1	2	4			1																															
13				3		1					1	2																														
14							4				1	2	9																													
15											1			11																												
16							3				3	2	6	1																												
17															11	11																										
18							1				1			5	3	1	2																									
19			1	1			2				1	1					3	1																								
20						1	2					1							9																							
21							1		1							1			1	9																						
22									5												11																					

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42

Table 7.1

Non-Redundant Preceding Elements

Element	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42
23																					10	1																				
24																					4	1	6																			
25			1	1		1	5				2	1				1																										
26																					6		2	2	10																	
27										1						7							2	1	5	5																
28							6				1														1		3															
29														3	3	1	1				5		1	3			2	1														
30							5	1			2	2																	2													
31																														11												
32																1									1					5	6											
33																					8	1								8		2										
34																													1				11									
35							8				2																		1													
36															1										1					8		1		2	9							
37							7				3												1																			
38																					6	2	1														9					
39																																					1	10				
40									1													1																	11			
41																					1																1	1	1	8		
42																																							1		11	
43		1										4									5				4	1			2	1												
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42

Table 7.1

(continued)

BIBLIOGRAPHY

1. Smallwood, R. D., Decision Structure for Teaching Machines, M. I. T. Press, 1962.

2. Green, B. F., "Computer Languages for Symbol Manipulations", IRE Transactions on Human Factors in Electronics, March, 1961.

3. McCarthy, Jetal, LISP 1.5 Programmers Manual, M. I. T. Press, 1962.

4. Smallwood, R. D., "Adaptive Teaching Machine Program for Miniature Geometry" (private publication).

5. Aitken, A. C., Determinants and Matrices, Ninth Edition Interscience Publishers, Inc., New York, 1956.

Security Classification

DOCUMENT CONTROL DATA - R&D

(Security classification of title, body of abstract and indexing annotation must be entered when the overall report is classified)

1. ORIGINATING ACTIVITY (Corporate author) Operations Research Center, Massachusetts Institute of Technology, Cambridge, Massachusetts	2a. REPORT SECURITY CLASSIFICATION Unclassified 2b. GROUP n/a

3. REPORT TITLE

Preliminary Research on the Taxonomy of Subject Matter.

4. DESCRIPTIVE NOTES (Type of report and inclusive dates)

None

5. AUTHOR(S) (Last name, first name, initial)

Roderburg, T. K., Cluck, H. D., Murrary, G. R. Jr.

6. REPORT DATE	7a. TOTAL NO. OF PAGES	7b. NO. OF REFS
November 1964	26	5

8a. CONTRACT OR GRANT NO. AF 19 (628)-2407 b. PROJECT NO. 7682 c. d.	9a. ORIGINATOR'S REPORT NUMBER(S) ESD-TDR-64-618 9b. OTHER REPORT NO(S) (Any other numbers that may be assigned this report) Technical Note #3

10. AVAILABILITY/LIMITATION NOTICES

Qualified requesters may obtain from DDC.
DDC release to OTS authorized.

11. SUPPLEMENTARY NOTES	12. SPONSORING MILITARY ACTIVITY
None	Decision Sciences Laboratory, Electronic Systems Division, L. G. Hanscom Field, Bedford, Massachusetts.

13. ABSTRACT

Preliminary research in the taxonomy of subject matter is reported. This work is part of a program of study aimed at developing computational methods useful in the preparation of educational matter for presentation by machine. Basic concepts of subject structure are defined. A language processing program that assists the classification of subject matter is described and its use illustrated. An experiment on the variations in subject structure as seen by different individuals is reported.

14. KEY WORDS	LINK A		LINK B		LINK C	
	ROLE	WT	ROLE	WT	ROLE	WT
1. Classification						
2. Data Processing Systems						
3. Teaching Machines						
4. Computers						
5. Language						
6. Machine Translation						
7. Programming						
8. Experimental Data						
9. Training Material						

INSTRUCTIONS

1. ORIGINATING ACTIVITY: Enter the name and address of the contractor, subcontractor, grantee, Department of Defense activity or other organization *(corporate author)* issuing the report.

2a. REPORT SECURITY CLASSIFICATION: Enter the overall security classification of the report. Indicate whether "Restricted Data" is included. Marking is to be in accordance with appropriate security regulations.

2b. GROUP: Automatic downgrading is specified in DoD Directive 5200.10 and Armed Forces Industrial Manual. Enter the group number. Also, when applicable, show that optional markings have been used for Group 3 and Group 4 as authorized.

3. REPORT TITLE: Enter the complete report title in all capital letters. Titles in all cases should be unclassified. If a meaningful title cannot be selected without classification, show title classification in all capitals in parenthesis immediately following the title.

4. DESCRIPTIVE NOTES: If appropriate, enter the type of report, e.g., interim, progress, summary, annual, or final. Give the inclusive dates when a specific reporting period is covered.

5. AUTHOR(S): Enter the name(s) of author(s) as shown on or in the report. Enter last name, first name, middle initial. If military, show rank and branch of service. The name of the principal author is an absolute minimum requirement.

6. REPORT DATE: Enter the date of the report as day, month, year; or month, year. If more than one date appears on the report, use date of publication.

7a. TOTAL NUMBER OF PAGES: The total page count should follow normal pagination procedures, i.e., enter the number of pages containing information.

7b. NUMBER OF REFERENCES: Enter the total number of references cited in the report.

8a. CONTRACT OR GRANT NUMBER: If appropriate, enter the applicable number of the contract or grant under which the report was written.

8b, 8c, & 8d. PROJECT NUMBER: Enter the appropriate military department identification, such as project number, subproject number, system numbers, task number, etc.

9a. ORIGINATOR'S REPORT NUMBER(S): Enter the official report number by which the document will be identified and controlled by the originating activity. This number must be unique to this report.

9b. OTHER REPORT NUMBER(S): If the report has been assigned any other report numbers *(either by the originator or by the sponsor)*, also enter this number(s).

10. AVAILABILITY/LIMITATION NOTICES: Enter any limitations on further dissemination of the report, other than those imposed by security classification, using standard statements such as:

(1) "Qualified requesters may obtain copies of this report from DDC."

(2) "Foreign announcement and dissemination of this report by DDC is not authorized."

(3) "U. S. Government agencies may obtain copies of this report directly from DDC. Other qualified DDC users shall request through
________________________________."

(4) "U. S. military agencies may obtain copies of this report directly from DDC. Other qualified users shall request through
________________________________."

(5) "All distribution of this report is controlled. Qualified DDC users shall request through
________________________________."

If the report has been furnished to the Office of Technical Services, Department of Commerce, for sale to the public, indicate this fact and enter the price, if known.

11. SUPPLEMENTARY NOTES: Use for additional explanatory notes.

12. SPONSORING MILITARY ACTIVITY: Enter the name of the departmental project office or laboratory sponsoring *(paying for)* the research and development. Include address.

13. ABSTRACT: Enter an abstract giving a brief and factual summary of the document indicative of the report, even though it may also appear elsewhere in the body of the technical report. If additional space is required, a continuation sheet shall be attached.

It is highly desirable that the abstract of classified reports be unclassified. Each paragraph of the abstract shall end with an indication of the military security classification of the information in the paragraph, represented as *(TS), (S), (C), or (U)*.

There is no limitation on the length of the abstract. However, the suggested length is from 150 to 225 words.

14. KEY WORDS: Key words are technically meaningful terms or short phrases that characterize a report and may be used as index entries for cataloging the report. Key words must be selected so that no security classification is required. Identifiers, such as equipment model designation, trade name, military project code name, geographic location, may be used as key words but will be followed by an indication of technical context. The assignment of links, rules, and weights is optional